МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ПЕРМСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

(ПНИПУ)

Факультет: Электротехнический

Кафедра: «Информационные технологии и автоматизированные системы» (ИТАС)

Направление: Разработка информационных систем (РИС)

**О Т Ч Ё Т**   
**о лабораторной работе №5**

Дисциплина: «Основы алгоритмизации и программирования»

Тема: «Функции и массивы»

Выполнил:

Студент группы РИС-24-3б

Суяргулов А.О.

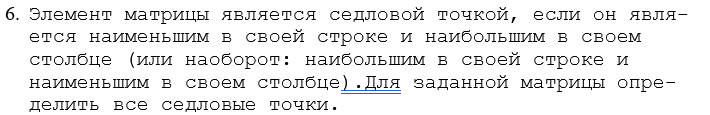
Проверил:

Доцент кафедры ИТАС

Полякова О.А.

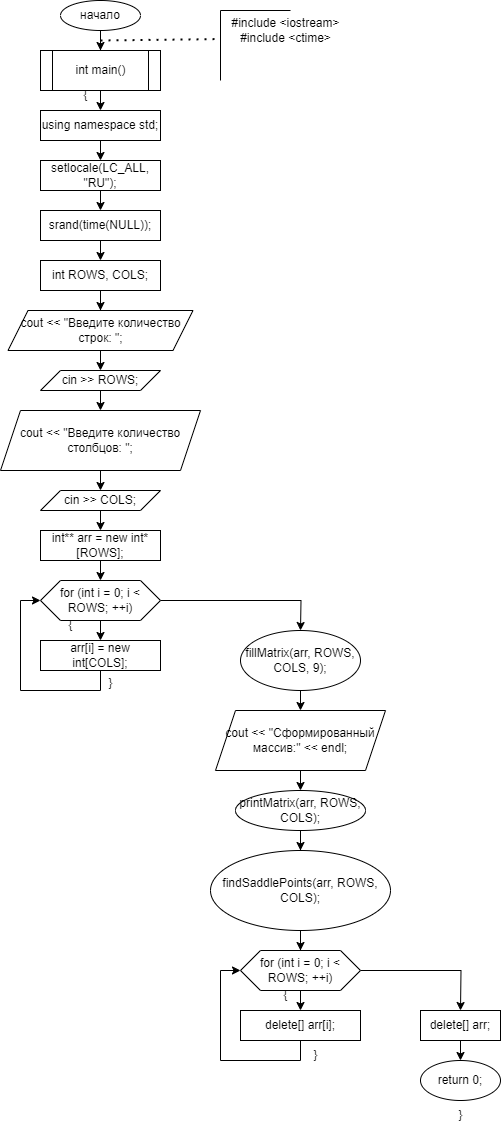
Пермь, 2025

ВВЕДЕНИЕ

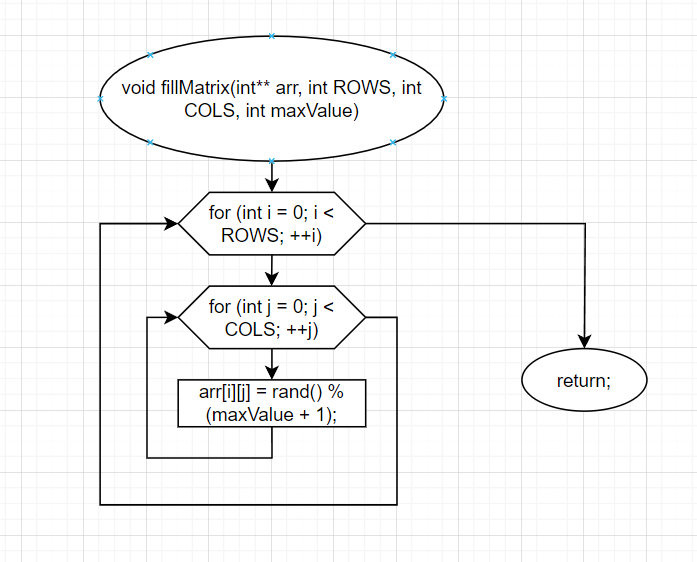


ОСНОВНАЯ ЧАСТЬ

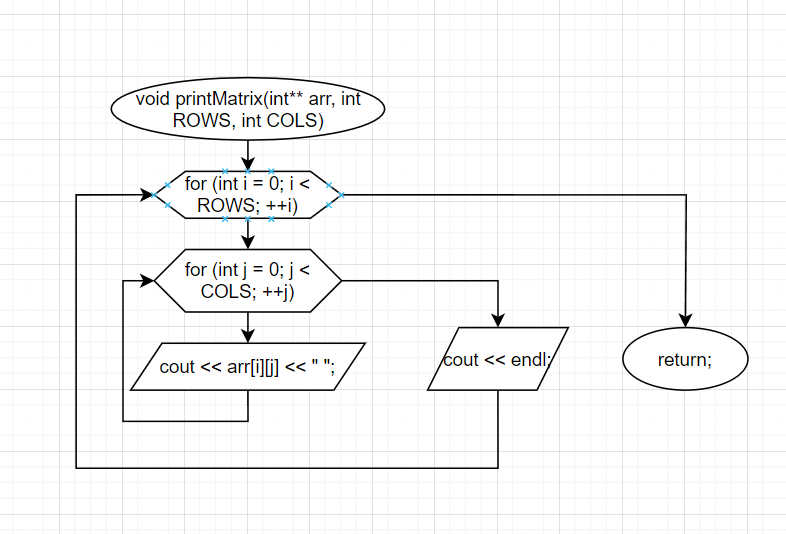
**Блок-схема:**



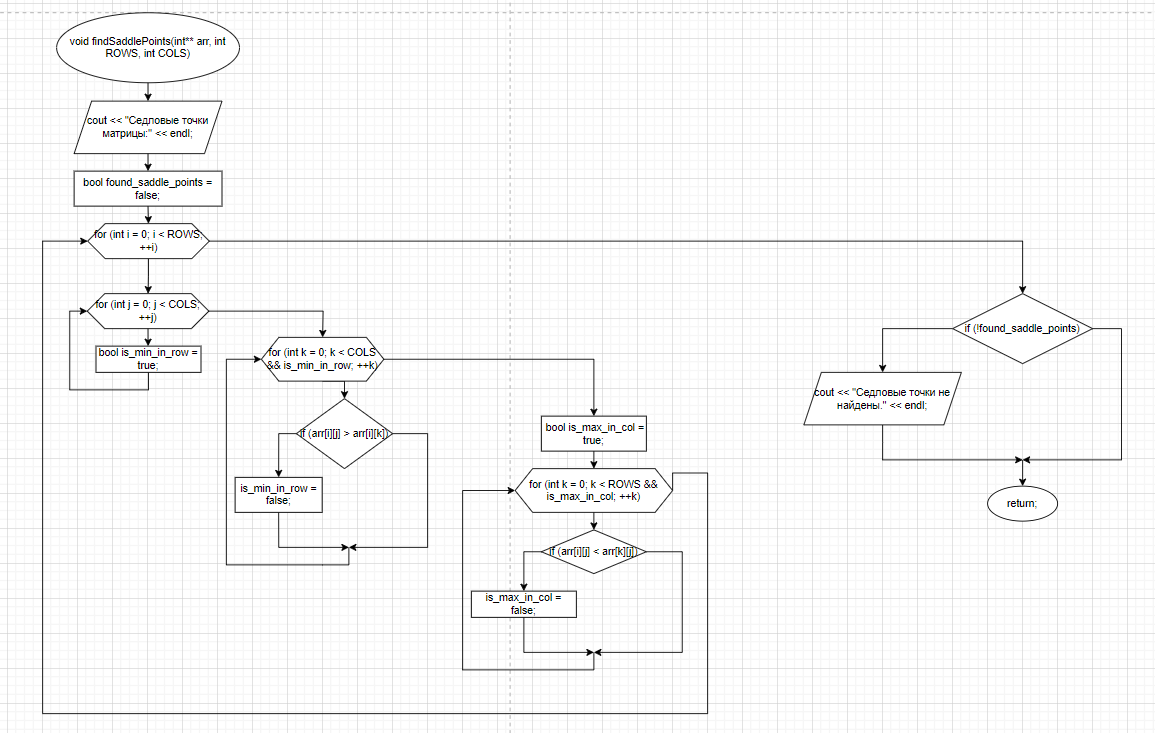
**Функция для заполнения матрицы:**

****

**Функция для печати массива:**

****

**Функция для поиска седловых точек:**

****

**Код:**

#include <iostream>

#include <ctime>

using namespace std;

// Функция для заполнения двумерного массива случайными числами

void fillMatrix(int\*\* arr, int ROWS, int COLS, int maxValue) {

for (int i = 0; i < ROWS; ++i) {

for (int j = 0; j < COLS; ++j) {

arr[i][j] = rand() % (maxValue + 1); // Заполняем случайными числами от 0 до maxValue

}

}

}

// Функция для вывода двумерного массива

void printMatrix(int\*\* arr, int ROWS, int COLS) {

for (int i = 0; i < ROWS; ++i) {

for (int j = 0; j < COLS; ++j) {

cout << arr[i][j] << " "; // Просто выводим число и пробел

}

cout << endl;

}

}

// Функция для поиска седловых точек

void findSaddlePoints(int\*\* arr, int ROWS, int COLS) {

cout << "Седловые точки матрицы:" << endl;

bool found\_saddle\_points = false;

for (int i = 0; i < ROWS; ++i) {

for (int j = 0; j < COLS; ++j) {

// Проверяем, является ли arr[i][j] наименьшим в своей строке и наибольшим в своем столбце

bool is\_min\_in\_row = true;

for (int k = 0; k < COLS && is\_min\_in\_row; ++k) { // Проверяем строку, пока is\_min\_in\_row истинно

if (arr[i][j] > arr[i][k]) {

is\_min\_in\_row = false;

}

}

bool is\_max\_in\_col = true;

for (int k = 0; k < ROWS && is\_max\_in\_col; ++k) { // Проверяем столбец, пока is\_max\_in\_col истинно

if (arr[i][j] < arr[k][j]) {

is\_max\_in\_col = false;

}

}

// Проверяем, является ли arr[i][j] наибольшим в своей строке и наименьшим в своем столбце

bool is\_max\_in\_row = true;

for (int k = 0; k < COLS && is\_max\_in\_row; ++k) { // Проверяем строку, пока is\_max\_in\_row истинно

if (arr[i][j] < arr[i][k]) {

is\_max\_in\_row = false;

}

}

bool is\_min\_in\_col = true;

for (int k = 0; k < ROWS && is\_min\_in\_col; ++k) { // Проверяем столбец, пока is\_min\_in\_col истинно

if (arr[i][j] > arr[k][j]) {

is\_min\_in\_col = false;

}

}

if ((is\_min\_in\_row && is\_max\_in\_col) || (is\_max\_in\_row && is\_min\_in\_col)) {

cout << "Элемент: " << arr[i][j] << " (строка: " << i + 1 << ", столбец: " << j + 1 << ")" << endl;

found\_saddle\_points = true;

}

}

}

if (!found\_saddle\_points) {

cout << "Седловые точки не найдены." << endl;

}

}

int main() {

setlocale(LC\_ALL, "RU");

srand(time(NULL)); // Инициализируем генератор случайных чисел

int ROWS, COLS;

cout << "Введите количество строк: ";

cin >> ROWS;

cout << "Введите количество столбцов: ";

cin >> COLS;

// Выделяем память для двумерного массива

int\*\* arr = new int\* [ROWS];

for (int i = 0; i < ROWS; ++i) {

arr[i] = new int[COLS];

}

// Заполняем массив случайными числами

fillMatrix(arr, ROWS, COLS, 9); // Заполняем числами от 0 до 9

// Выводим массив

cout << "Сформированный массив:" << endl;

printMatrix(arr, ROWS, COLS);

// Ищем седловые точки

findSaddlePoints(arr, ROWS, COLS);

// Освобождаем память

for (int i = 0; i < ROWS; ++i) {

delete[] arr[i];

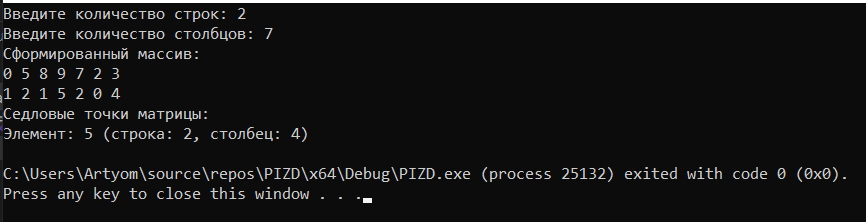
}

delete[] arr;

return 0;

}

**Результат работы кода:**



**Ссылка на GitHub:**

<https://github.com/Drazda1488/LAB_5>